AMENDMENT

In the Claims

Please amend claims 9, 11, 15, 17, 27, 29, and 32.

Please cancel claims 1-8, 10, 12-14, 16, 18-26, 28, 30, 31, and 33-36 without prejudice.

1. (Cancelled)

2. (Cancelled)

3. (Cancelled)

4. (Cancelled)

5. (Cancelled)

6. (Cancelled)

7. (Cancelled)

8. (Cancelled)

9. (Currently Amended) A machine-readable medium that provides executable firmware instructions which, when executed by a processor in a computer system having a native environment that executes in physical mode, cause the processor to perform operations comprising:

implementing an extensible firmware framework via which firmware modules are loaded during a pre-boot phase of the computer system

implementing a firmware-based virtual machine monitor (VMM) upon [[a]] the computing system ~~having a native environment that executes in physical mode~~; ~~and~~

emulating legacy hardware components that are not present in the native environment using the VMM to provide support for legacy code running on the computer system; and authenticating, via the VMM, a firmware module that is loaded during the pre-boot phase by comparing a digital signature provided with the firmware module with digital signatures stored in secure storage that is accessible to the VMM.

10. (Cancelled)

11. (Currently Amended) The machine-readable medium of claim [[10]] 9, wherein the VMM provides at least one of PC/AT hardware emulation and PC/AT environment emulation.

12. (Cancelled)

13. (Cancelled)

14. (Cancelled)

15. (Currently Amended) An apparatus comprising:

a computing system having a native execution environment that executes in physical mode, the computer system including an extensible firmware framework via which firmware modules are loaded during a pre-boot phase of the computer system; and

a virtual machine monitor implemented thereon, the virtual machine monitor emulating legacy hardware components that are not present in the native environment to provide support for legacy code to run on the computer system, the virtual machine monitor further authenticating a firmware module loaded during the pre-boot phase by comparing a digital signature provided with the firmware module with digital signatures stored in secure storage accessible to the VMM.

16. (Cancelled)

17. (Currently Amended) The apparatus of claim [[16]] 15, wherein the VMM provides at least one of PC/AT hardware emulation and PC/AT environment emulation.

18. (Cancelled)
19. (Cancelled)
20. (Cancelled)
21. (Cancelled)
22. (Cancelled)
23. (Cancelled)
24. (Cancelled)
25. (Cancelled)
26. (Cancelled)

27. (Currently Amended) A method, comprising:

implementing a virtual machine monitor (VMM) during the pre-boot phase of a computer system; ~~and~~

~~authenticating an Extensible Firmware Interface (EFI) firmware module using the VMM.~~

storing digital signatures of valid firmware modules in secure storage; and

authenticating a firmware module via the VMM by comparing a digital signature provided with the firmware module to the digital signatures in the secure storage.

28 (Cancelled)

29 (Currently Amended) The method of claim [[28]] 27, further comprising:

maintaining an attestation log via the VMM identifying firmware modules that have been loaded and authenticated by the VMM.

30. (Cancelled)

31. (Cancelled)

32. (Currently Amended) The machine-readable medium of claim [[30]] 9, wherein execution of the instructions cause further operations to be performed, comprising:

maintaining an attestation log via the VMM identifying firmware modules that have been loaded and authenticated by the VMM.

33. (Cancelled)

34. (Cancelled)

35. (Cancelled)

36. (Cancelled)

37. (Previously Presented) The apparatus of claim 15, wherein the VMM performs further operations, including:

enabling a legacy option ROM (read-only memory) to run and effect its input/output (I/O) services; and

translating the results of the I/O services into a native API (application program interface).